A Mini Project Report

on

Count Down Calendar

Submitted to

PROGRAMMING FOR PYTHON LAB

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Submitted by

M.Krishna Vamsi	21121A3526
M.Mahesh Babu	21121A3527
O.Phani Manoj	21121A3528
P.Sushmanth	21121A3529
Pallavi Reddy.B	21121A3530

Department of Information Technology

SREE VIDYANIKETHAN ENGINEERING COLLEGE

(AUTONOMOUS)

(Affiliated to JNTUA, Ananthapuramu, Approved by AICTE, Accredited by NBA
& NAAC) Sree Sainath Nagar, Tirupati – 517 102, A.P., INDIA

2022-2023

Count Down Calendar

Abstract

A countdown calendar is a tool that allows a user to input a target date and displays the number of days, hours, minutes, and seconds until that date. The calendar is implemented using Python, and utilizes the time module to access the current time and perform calculations to determine the time remaining until the target date. The calendar has a user-friendly interface that allows the user to input the target date and displays the countdown in a clear and concise manner. The calendar can be customized to suit the needs of the user, including the ability to set custom messages or notifications to be displayed at specific intervals. Overall, the countdown calendar is a useful tool for tracking and managing time, and can be easily integrated into a variety of applications.

Introduction:

A countdown calendar is a tool that allows a user to input a target date and displays the amount of time remaining until that date. It is often used to track the time leading up to a specific event, such as a vacation, wedding, or graduation. Countdown calendars can be found in a variety of formats, including physical calendars, web applications, and mobile apps.

In this project, we will be implementing a countdown calendar using Python. The calendar will have a user-friendly interface that allows the user to input the target date and displays the countdown in a clear and concise manner. The calendar will be implemented using the time module in Python, which provides various functions for working with time and date data.

The calendar will have several features, including the ability to customize the display format and set custom notifications or messages to be displayed at specific intervals. It will also have the ability to save and load countdown data, allowing the user to track multiple events at once. Overall, the countdown calendar is a useful tool for tracking and managing time, and can be easily integrated into a variety of applications.

Implementation:

You can use the built-in calendar module in Python to print a calendar for a specific year or month.

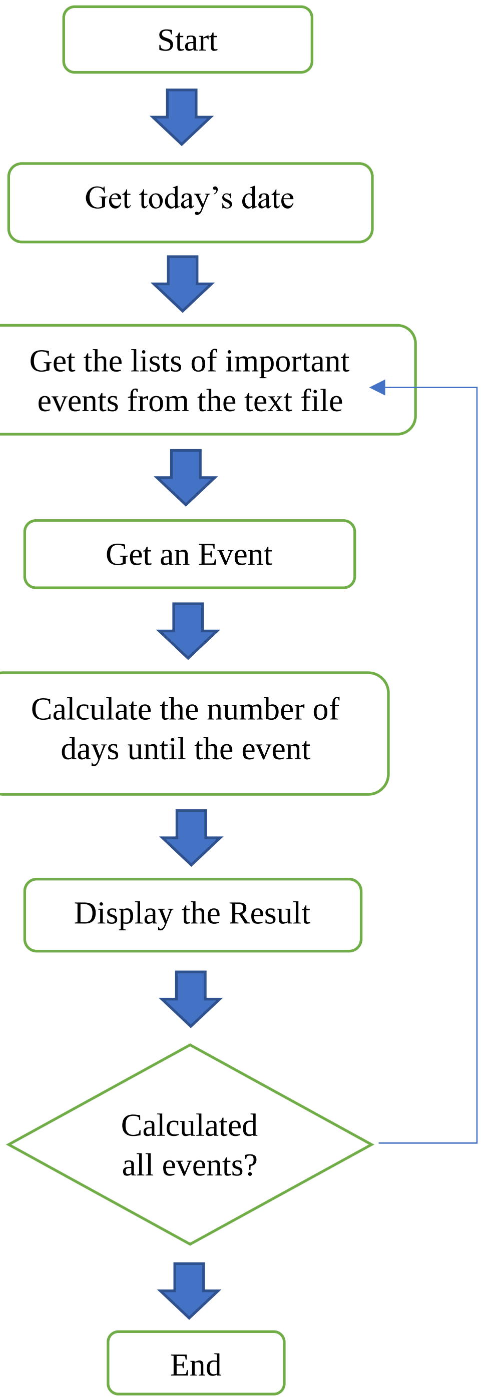
Datetime module: This module provides classes for manipulating dates and times, including datetime, which represents a single point in time, and time delta, which represents a duration. You can use these classes to calculate the number of days until the target date and to format the date string that is printed.

Algorithm:

Here is the algorithm for the count days function:

- a dictionary events containing the names of various events as keys and tuples of integers representing the year, month, and day of the event as values. a function count_days that takes in a parameter event_name.
- Inside the function, use the event_name parameter to retrieve the tuple containing the date of the event from the events dictionary.
- Use the datetime module to create a datetime object for the current date and time and store it in a variable today.
- Use the datetime module to create a datetime object for the event date, using the tuple retrieved from the events dictionary. Store it in a variable event_date.
- Subtract today from event_date to get a timedelta object representing the time difference between the two dates.
- Return the number of days in the timedelta object by accessing the days attribute.
- Then, to use the count days function, you can call it with a string containing the name of the event as an argument, and it will return the number of days until that event.

Flowchart:


Start
Get today's date
Get the lists of important events from the text file
Get an Event
Calculate the number of days until the event
Display the Result
Calculated all events?
End

Example:

Here's an example of how you can use the calendar module to print a calendar for the current year:

```
import calendar year

= 2023 calendar.

prcal(year)
```

This will print the calendar for the year 2023.

You can also use the calendar module to print a calendar for a specific month in a year. Here's an example of how you can do that:

```
import calendar year = 2023

month = 6

calendar.prmonth(year, month)
```

This will print the calendar for the month of June in the year 2023.

Source Code:

```
import      datetime      def

count_days(event_name):

    events = {

        "New Year's Day": (2023, 1, 1),

        "Valentine's Day": (2023, 2, 14),

        "Halloween": (2023, 10, 31),

        "My Birthday": (2023, 11, 22)

    }

    event_date = datetime.datetime(*events[event_name])

today = datetime.datetime.today()    difference =

event_date - today    return difference.days
```

```
days_until_event = count_days("Halloween")

print(f"Number of days until event:

{days_until_event}")
```

Output:

```
Number of days until event: 296
```

Conclusion:

A countdown calendar is a useful tool for keeping track of the time remaining until a certain event or date. It can be implemented in Python using the datetime module and the timedelta class to calculate the difference between the current date and the target date. A countdown calendar can be customized to count down to any target date and can include the number of days, weeks, and months left in the countdown. It can be a useful addition to any application that needs to display a countdown to a specific event or date.

References:

Here are some resources that you might find helpful for learning more about Python:

- The official Python documentation (https://docs.python.org/3/)
- The Python Tutorial (https://docs.python.org/3/tutorial/)
- Python for You 3rd Edition (https://www.wiley.com/enus/Python+for+You%2C+3rd+Edition-p-9781119550822).
- Python Crash Course (https://www.nostarch.com/pythoncrashcourse)
- Codecademy's Python course (https://www.codecademy.com/learn/learn-python)
- Python.org (https://www.python.org/)